Good Neighbors Make Good Senses: Exploiting Distributional Similarity for Unsupervised WSD

Good Neighbors Make Good Senses: Exploiting Distributional Similarity for Unsupervised WSD

Samuel Brody
School of Informatics
University of Edinburgh
s.brody@sms.ed.ac.uk

Mirella Lapata
School of Informatics
University of Edinburgh
mlap@inf.ed.ac.uk

Abstract

We present an automatic method for sense-labeling of text in an unsupervised manner. The method makes use of distributionally similar words to derive an automatically labeled training set, which is then used to train a standard supervised classifier for distinguishing word senses. Experimental results on the Senseval-2 and Senseval-3 datasets show that our approach yields significant improvements over state-of-the-art unsupervised methods, and is competitive with supervised ones, while eliminating the annotation cost.

1 Introduction

Word sense disambiguation (WSD), the task of identifying the intended meaning (sense) of words in context, is a long-standing problem in Natural Language Processing. Sense disambiguation is often characterized as an intermediate task, which is not an end in itself, but has the potential to improve many applications. Examples include summarization (Barzilay and Elhadad, 1997), question answering (Ramakrishnan et al., 2003) and machine translation (Chan and Ng, 2007).

WSD is commonly treated as a supervised classification task. Assuming we have access to data that has been hand-labeled with correct word senses, we can train a classifier to assign senses to unseen words in context. While this approach often achieves high accuracy, adequately large sense labeled data sets are unfortunately difficult to obtain. For many words, domains, languages, and sense inventories they are unavailable, and in most cases it is unreasonable to expect to acquire them. Ng (1997) estimates that a high accuracy domain-independent system for WSD would probably need a corpus of about 3.2 million sense tagged words. At a throughput of one word per minute (Edmonds, 2000), this would require about 27 person-years of human annotation effort.

SemCor (Fellbaum, 1998) is one of the few corpora that have been manually annotated for all words — it contains sense labels for 23,346 lemmas. In spite of being widely used, SemCor contains too few tagged instances for the majority of polysemous words (typically fewer than 10 each). Supervised methods require much larger data sets than this to perform adequately.

The problem of obtaining sufficient labeled data, often referred to as the *data acquisition bottleneck*, creates a significant barrier to the use of supervised WSD methods in real world applications. In this work we wish to take advantage of the high accuracy and strong capabilities of supervised methods, while eliminating the need for human annotation of training data. Our approach exploits a sense inventory such as WordNet (Fellbaum, 1998) and corpus data to *automatically* create a collection of sense labeled instances which can subsequently serve to train any supervised classifier. The key premise of our work is that a word's senses can be broadly described by semantically related words. So, rather than laboriously annotating all instances of a polysemous word with its senses, we collect instances of its related words and treat them as sense labels for the target word. The method is inexpensive, language-independent, and can be used to create large sense-labeled data without human intervention. Our results demonstrate significant improvements over state-of-the-art unsupervised methods that do not make use of hand-labeled annotations.

In the following section we provide an overview

Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008), pages 65–72
Manchester, August 2008

of existing work on unsupervised WSD. Section 3 introduces our method for automatically creating sense annotations. We present our evaluation framework in Section 4 and results in Section 5.

2 Related Work

The data requirements for supervised WSD and the current paucity of suitably annotated corpora for many languages and text genres, has sparked considerable interest in unsupervised methods. These typically come in two flavors: (1) developing algorithms that assign word senses without relying on a sense-labeled corpus (Lesk, 1986; Galley and McKeown, 2003) and (2) making use of pseudo-labels, i.e., labelled data that has not been specifically annotated for sense disambiguation purposes but contains some form of sense distinctions (Gale et al., 1992; Leacock et al., 1998). We briefly discuss representative examples of both approaches, with a bias to those closely related to our own work.

Unsupervised Algorithms One of the first approaches to unsupervised WSD, and the foundation of many algorithms to come, was originally introduced by Lesk (1986). The method assigns a sense to a target ambiguous word by comparing the dictionary definitions of each of its senses with the words in the surrounding context. The sense whose definition has the highest overlap (i.e., words in common) with the context is assumed to be the correct one. Despite its simplicity, the algorithm provides a good baseline for comparison. Coverage can be increased by augmenting the dictionary definition (gloss) of each sense with the glosses of related words and senses (Banerjee and Pedersen, 2003).

Although most algorithms disambiguate word senses in context, McCarthy et al. (2004) propose a method that does not rely on contextual cues. Their algorithm capitalizes on the fact that the distribution of word senses is highly skewed. A large number of frequent words is often associated with one dominant sense. Indeed, current supervised methods rarely outperform the simple heuristic of choosing the most common sense in the training data (henceforth "the first sense heuristic"), despite taking local context into account. Rather than obtaining the first sense via annotating word senses manually, McCarthy et al. propose to acquire first senses automatically and use them for disambiguation. Thus, by design, their algorithm assigns the same sense to all instances of a polysemous word.

Their approach is based on the observation that distributionally similar neighbors often provide cues about a word's senses. Assuming that a set of neighbors is available, the algorithm quantifies the degree of similarity between the neighbors and the sense descriptions of the polysemous word. The sense with the highest overall similarity is the first sense. Specifically, the approach makes use of two similarity measures which complement each other and provide a large amount of data regarding the word senses. Distributional similarity indicates the similarity between words in the distributional feature space, whereas WordNet similarity in the 'semantic' space, is used to discover which sense of the ambiguous word is used in the corpus, and causing the distributional similarity.

Pseudo-labels as Training Instances Gale et al. (1992) pioneered the use of parallel corpora as a source of sense-tagged data. Their key insight is that different translations of an ambiguous word can serve to distinguish its senses. Ng et al. (2003) extend this approach further and demonstrate that it is feasible for large scale WSD. They gather examples from English-Chinese parallel corpora and use automatic word alignment as a means of obtaining a translation dictionary. Translations are next assigned to senses of English ambiguous words. English instances corresponding to these translations serve as training data.

It has become common to use related words from a dictionary to learn contextual cues for WSD (Mihalcea, 2002). Perhaps the first incarnation of this idea is found in Leacock et al. (1998), who describe a system for acquiring topical contexts that can be used to distinguish between senses. For each sense, related monosemous words are extracted from WordNet using the various relationship connections between sense entries (e.g., hyponymy, hypernymy). Their system then queries the Web with these related words. The contexts surrounding the relatives of a specific sense are presumed to be indicators of that sense, and used for disambiguation. A similar idea, proposed by Yarowsky (1992), is to use a thesaurus and acquire informative contexts from words in the same category as the target.

Our own work uses insights gained from unsupervised methods with the aim of creating large datasets of sense-labeled instances without explicit manual coding. Unlike Ng et al. (2003) our algorithm works on monolingual corpora, which are

much more abundant than parallel ones, and is fully automatic. In their approach translations and their English senses must be associated manually. Similarly to McCarthy et al. (2004), we assume that words related to the target word are useful indicators of its senses. Importantly, our method disambiguates words in context and is able to assign additional senses, besides the first one.

3 Method

As discussed earlier, our aim is to alleviate the need for manual annotation by creating a large dataset labeled with word senses without human intervention. This dataset can be subsequently used by any supervised machine learning algorithm. We assume here that we have access to a corpus and a sense inventory. We first obtain a list of words that are semantically related to our target word. In the remainder of this paper we use the term "neighbors" to refer to these words. Next, we separate the neighbors into sense-specific groups. Finally, we replace the occurrences of each neighbor in our corpus with an instance of the target word, labeled with the matching sense for that neighbor. The procedure has two important steps: (1) acquiring neighbors and (2) associating them with appropriate senses. We describe our implementation of each stage in more detail below.

Neighbor Acquisition Considerable latitude is allowed in specifying appropriate neighbors for the target word. Broadly speaking, the neighbors can be extracted from a corpus or from a semantic resource, for example the dictionary providing the sense inventory. A wealth of algorithms have been proposed in the literature for acquiring distributional neighbors from a corpus (see Weeds (2003) for an overview). They differ as to which features they consider and how they use the distributional statistics to calculate similarity.

Lin's (1998) information-theoretic similarity measure is commonly used in lexicon acquisition tasks and has demonstrated good performance in unsupervised WSD (McCarthy et al., 2004). It operates over dependency relations. A word w is described by a set $T(w)$ of co-occurrence triplets $< w, r, w' >$, which can be viewed as a sparsely represented feature vector, where r represents the type of relation (e.g., *object-of*, *subject-of*, *modified-by*) between w and its dependent w'. The similarity between w_1 and w_2 is then defined as:

$$\frac{\sum_{(r,w)\in T(w_1)\cap T(w_2)} I(w_1,r,w)+I(w_2,r,w)}{\sum_{(r,w)\in T(w_1)} I(w_1,r,w)+\sum_{(r,w)\in T(w_2)} I(w_2,r,w)}$$

where $I(w,r,w')$ is the *information value* of w with regard to (r,w'), defined as:

$$I(w,r,w') = \log \frac{count(w,r,w')\cdot count(r)}{count(*,r,w')\cdot count(w,r,*)}$$

The measure is used to estimate the pairwise similarity between the target word and all other words in the corpus (with the same part of speech); the k words most similar to the target are selected as its neighbors.

A potential caveat with Lin's (1998) distributional similarity measure is its reliance on syntactic information for obtaining dependency relations. Parsing resources may not be available for all languages or domains. An alternative is to use a measure of distributional similarity which considers word collocation statistics and therefore does not require a syntactic parser (see Weeds (2003)).

As mentioned earlier, it is also possible to obtain neighbors simply by consulting a semantic dictionary. In WordNet, for example, we can assume that WordNet relations, (e.g., hypernymy, hyponymy, synonymy) indicate words which are semantic neighbors. An advantage of using distributional neighbors is that they reflect the characteristics of the corpus we wish to disambiguate and are potentially better suited for capturing sense differences across genres and domains, whereas dictionary-based neighbors are corpus-invariant.

Associating Neighbors with Senses If the neighbors are extracted from WordNet, it is not necessary to associate them with their senses as they are already assigned a specific sense. Distributional similarity methods, however, do not provide a way to distinguish which neighbors pertain to each sense of the target. For that purpose, we adapt a method proposed by McCarthy et al. (2004). Specifically, for each acquired neighbor, we choose the sense of the target which gives the highest semantic similarity score to *any* sense of the neighbor. There are a large number of semantic similarity measures to choose from (see Budanitsky and Hirst (2001) for an overview). We use Lesk's measure as modified by Banerjee and Pedersen (2003) for two reasons. First, it has

been shown to perform well in the related task of predominant sense detection (McCarthy et al., 2004). Second, it has the advantage of relying only upon the sense definitions, rather than the complex graph structure which is unique to WordNet. This makes the method more suitable for use with other sense inventories.

Note that unlike McCarthy et al. (2004), we are associating neighbors with senses, rather than merely trying to detect the predominant sense, and therefore we require more precision in our selection. When it is unclear which sense of the target word is most similar to a given neighbor (when the scores of two or more senses are close together), that neighbor is discarded.

As an example, consider the word *sense*, which has four meanings[1] in WordNet: (1) a general conscious **awareness** (e.g., *a sense of security*), (2) the **meaning** of a word (e.g., *the dictionary gave several senses for the word*), (3) sound practical **judgment** (e.g., *I can't see the sense in doing it now*), and (4) a natural appreciation or **ability** (e.g., *keen musical sense*). On the British National Corpus (BNC), using Lin's (1998) similarity method, we retrieve the following neighbors for the first and second sense, respectively:

1. awareness, feeling, instinct, enthusiasm, sensation, vision, tradition, consciousness, anger, panic, loyalty
2. emotion, belief, meaning, manner, necessity, tension, motivation

No neighbors are associated with the last two senses, indicating that they are not prevalent enough in the BNC to be detected by this method.

Once sense-specific neighbors are acquired, the next stage is to replace all instances of the neighbors in the corpus with the target ambiguous word labeled with the appropriate sense. For example, when encountering the sentence *"... attempt to state the **meaning** of a word"*, our method would automatically transform this to *"... attempt to state the **sense** (s#2) of a word."* These *pseudo-labeled* instances comprise the training instances we provide to our machine learning algorithms.

4 Experimental Setup

We evaluated the performance of our approach on benchmark datasets. In this section we give details regarding our training and test data, and describe the features and machine learners we employed. Finally, we discuss the methods to which we compare our approach.

4.1 Data

Our experiments use a subset of the data provided for the English lexical sample task in the Senseval 2 (Preiss and Yarowsky, 2001) and Senseval 3 (Mihalcea and Edmonds, 2004) evaluation exercises. Since our method does not require hand tagged training data, we merged the provided training and test data into a single test set.

As a proof of concept we focus on the disambiguation of nouns, since they constitute the largest portion of content words (50% in the BNC). In addition, WordNet, which is our semantic resource and point of comparison, has a wide coverage of nouns. Also, for many tasks and applications (e.g., web queries) nouns are the most frequently encountered part-of-speech (Jansen et al., 2000). We made use of the coarse-grained sense groupings provided for both Senseval datasets. For many applications (e.g., information retrieval) coarsely defined senses are more useful (see Snow et al. (2007) for discussion).

Our training data was created from the BNC using different ways of obtaining the neighbors of the target word. As described in Section 3 we retrieved neighbors using Lin's (1998) similarity measure on a RASP parsed (Briscoe and Carroll, 2002) version of the BNC. We used subject and object dependencies, as well as adjective and noun modifier dependencies. We also created training data sets using collocational neighbors. Specifically, using the InfoMap toolkit[2], we constructed vector-based representations for individual words from the BNC using a term-document matrix and the cosine similarity measure. Vectors were initially constructed with 1,000 dimensions, the most frequent content words. The space was reduced to 100 dimensions with singular value decomposition. Finally, we also extracted neighbors from WordNet using first-order and sibling relations (i.e., hyponyms of the same hypernym). A problem often encountered when using dictionary-based neighbors is that they are themselves polysemous, and the related sense is often not the most prominent one in the corpus, which leads to noisy data. We therefore experimented with using *all* neighbors for a given word

[1] We are using the coarse-grained representation according to Senseval 2 annotators. The sense definitions are simplified for the sake of brevity.

[2] http://infomap.stanford.edu/

"The philosophical explanation of authority is not an attempt to state the **sense** of a word."	
Contextual features	
±10 words	explanation, of, authority, be, ...
±5 words	an, attempt, to, state, of, a, ...
Collocational features	
-2/+0 *n*-gram	_state_the_X
-1/+1 *n*-gram	_the_X_of
-0/+2 *n*-gram	_X_of_a
-2/+0 POS *n*-gram	_Verb_Det_X
-1/+1 POS *n*-gram	_Det_X_Prep
-0/+2 POS *n*-gram	_X_Prep_Det
Syntactic features	
Object of Verb	obj_of_state

Table 1: Example sentence and extracted features for the word *sense*; X denotes the target word.

or only those which are *monosemous* and hopefully less noisy. In all cases we used 50 neighbors, the most similar nouns to the target.

4.2 Features

We used a rich feature space based on lemmas, part-of-speech (POS) tags and a variety of positional and syntactic relationships of the target word capturing both immediate local context and wider context. These feature types have been widely used in WSD algorithms (see Lee and Ng (2002) for an evaluation of their effectiveness). Their use is illustrated on a sample English sentence for the target word *sense* in Table 1.

4.3 Supervised Classifiers

One of our evaluation goals was to examine the effect of our training-data creation procedure on different types of classifiers and determine which ones are most suited for use with our method. We therefore chose three supervised classifiers (support vector machines, maximum entropy, and label propagation) which are based on different learning paradigms and have shown competitive performance in WSD (Niu et al., 2005; Preiss and Yarowsky, 2001; Mihalcea and Edmonds, 2004). We summarize below their main characteristics and differences.

Support Vector Machines SVMs model classification as the problem of finding a separating hyperplane in a high dimensional vector space. They focus on differentiating between the most problematic cases — instances which are close to each other in the high dimensional space, but have different labels. They are discriminative, rather than generative, and do not explicitly model the classes. SVMs have been applied successfully in many NLP tasks. We used the multi-class bound-constrained support vector classification (SVC) version of SVM described in Hsu and Lin (2001) and implemented in the BSVM package[3]. All parameters were set to their default values with the exception of the misclassification penalty, which was set to a high value (1,000) to penalize labeling all instances with the most frequent sense.

Maximum Entropy Model Maximum entropy-based classifiers are a common alternative to other probabilistic classifiers, such as Naive Bayes, and have received much interest in various NLP tasks ranging from part-of-speech tagging to parsing and text classification. They represent a probabilistic, global constrained approach. They assume a uniform, zero-knowledge model, under the constraints of the training dataset. The classifier finds the (unique) maximal entropy model which conforms to the expected feature distribution of the training data. In our experiments, we used Megam[4] a publicly available maximum entropy classifier (Daumé III, 2004) with the default parameters.

Label Propagation The basic Label Propagation algorithm (Zhu and Ghahramani, 2002) represents labeled and unlabeled instances as nodes in an undirected graph with weighted edges. Initially only the known data nodes are labeled. The goal is to propagate labels from labeled to unlabeled points along the weighted edges. The weights are based on distance in a high-dimensional space. At each iteration, only the original labels are fixed, whereas the propagated labels are "soft", and may change in subsequent iterations. This property allows the final labeling to be affected by more distant labels, that have propagated further, and gives the algorithm a global aspect. We used SemiL[5], a publicly available implementation of label propagation (all parameters were set to default values).

4.4 Comparison with State-of-the-art

As an upper bound, we considered the accuracy of our classifiers when trained on the manually-labeled Senseval data (using the same experimental settings and 5-fold crossvalidation). This can be used to estimate the expected decrease in accuracy caused solely by the use of our automatic sense labeling method. We also compared our approach to other unsupervised ones. These include McCarthy

[3] http://www.csie.ntu.edu.tw/~cjlin/bsvm/
[4] http://www.isi.edu/~hdaume/megam/index.html
[5] http://www.engineers.auckland.ac.nz/~vkec001

et al.'s (2004) method for inferring the predominant sense and Lesk's (1986) algorithm. We modified the latter slightly so as to increase its coverage and used McCarthy et al.'s first sense heuristic to disambiguate unknown instances where no overlap was found. For McCarthy et al. we used parameters they report as optimal.

5 Results

The evaluation of our method was motivated by three questions: (1) How do different choices in constructing the pseudo-labeled training data affect WSD performance? Here, we would like to assess whether the origin of the target word neighbors (e.g., from a corpus or dictionary) matters. (2) What is the degree of noise and subsequent loss in accuracy incurred by our method? (3) How does the proposed approach compare against other unsupervised methods? In particular, we are interested to find out whether we outperform McCarthy et al.'s (2004) related method for predominant sense detection.

5.1 The Choice of Neighbors

Our results are summarized in Table 2. We report accuracy (rather than F-score) since all algorithms labeled all instances. The three center columns present our results with the automatically constructed training sets.

The best accuracies are observed when the labels are created from distributionally similar words using Lin's (1998) dependency-based similarity measure (Depend). We observe a small decrease in performance (within the range of 2%–4%) when using collocational neighbors without any syntactic information.[6] Using the neighbors provided by WordNet leads to worse results than using distributional neighbors. The differences in performance are significant[7] ($p < 0.01$) on both Senseval datasets for all classifiers and for both WordNet configurations, i.e., using all neighbors (AllWN) vs. monosemous ones (MonoWN).

This result may seem counterintuitive since neighbors provided by a semantic resource are based on expert knowledge and are often more accurate than those obtained automatically. However, semantic resources like WordNet are designed to be as general as possible without a specific corpus or domain in mind. They will therefore provide related words for all senses, even rare ones, which may not appear in our chosen corpus. Distributional methods, on the other hand, are anchored in the corpus. The extracted neighbors are usually relevant and representative of the corpus. Another drawback of resource-based neighbors is that they often do not share local behavior, i.e., they do not appear in the same immediate local context and do not share the same syntax. For this reason, the useful information that can be extracted from their contexts tends to be topical (e.g., words that are indicative of the domain), rather than local (e.g., grammatical dependencies). Topical information is mostly useful when the difference between senses can be attributed to a specific domain. However, for many words and senses, this is not the case (Leacock et al., 1998).

5.2 Comparison against Manual Labels

The rightmost column of Table 2 shows the accuracy of our classifiers when these are trained on the manually annotated Senseval datasets. In general, all algorithms exhibit a similar level of performance when trained on hand-coded data, with slightly lower scores for Senseval 3. On Senseval 2, the SVM is significantly better than the other two classifiers ($p < 0.01$). On Senseval 3, label propagation is significantly worse than the others ($p < 0.01$). The results shown here do not represent the highest achievable performance in a supervised setting, but rather those obtained without extensive parameter tuning. The best performing systems on coarse-grained nouns in Senseval 2 and 3 (Preiss and Yarowsky, 2001; Mihalcea and Edmonds, 2004) achieved approximately 76% and 80%, respectively. Besides being more finely tuned, these systems employed more sophisticated learning paradigms (e.g., ensemble learning).

When we compare the results from the manually labeled data to those achieved with the distributional neighbors, we can see that use of our pseudo-labels results in accuracies that are approximately 8–10% lower. Since the results were achieved using the same feature set and classifier settings, the comparison provides an estimate of the expected decrease in accuracy due only to our unsupervised tagging method. With more detailed feature engineering and more sophisticated machine learning methods, we could probably improve our classifiers' performance on the automatically labeled dataset. Also note that improvements in supervised methods can be expected to automatically translate to improvements in unsupervised

[6] We omit these results from the table for brevity.

[7] Throughout, we report significance using a χ^2 test.

Senseval 2	AllWN	MonoWN	Depend	Manual
SVM	48.12	53.29	64.38	72.52
MaxEnt	40.93	52.11	62.32	71.91
LP	42.67	49.54	63.32	69.28
McCarthy	59.98			
Lesk	48.12			

Senseval 3	AllWN	MonoWN	Depend	Manual
SVM	53.16	46.32	57.47	71.22
MaxEnt	49.67	44.85	57.35	71.75
LP	47.41	43.60	60.60	67.57
McCarthy	57.14			
Lesk	48.66			

Table 2: Accuracy (%) on Senseval 2 and 3 lexical samples. Support vector machines (SVM), maximum entropy (MaxEnt) and label propagation (LP) are trained on automatically and manually labeled data sets

WSD using our method.

Interestingly, label propagation performed relatively poorly on the manually labeled data. However, it ranks highly when using the automatic labels. This may be due to the fact that LP is the only algorithm that does not separate the training and test set (it is principally a semi-supervised method), allowing the properties of both to influence the structure of the resulting graph. Since the instances in the training data are not actual occurrences of the target word, it is important to learn which instances in the training set are closest to a given instance in the test set. The two other algorithms only attempt to distinguish between classes in the training set.

5.3 Other Unsupervised Methods

As shown in Table 2 our classifiers are significantly better than Lesk on both Senseval datasets ($p < 0.01$). They also significantly outperform the automatically acquired predominant sense (McCarthy) on Senseval 2 (for the Maximum Entropy classifier, the difference is significant at $p < 0.05$). On Senseval 3, all classifiers quantitatively outperform the first sense heuristic, but the difference is statistically significant only for label propagation ($p < 0.01$). The differences in performance on the two datasets can be explained by analyzing their sense distributions. Senseval 3 has a higher level of ambiguity (4.35 senses per word, on average, compared to 3.28 for Senseval 2), and is therefore a more difficult dataset. Although Senseval 3 has a slightly lower percentage of first sense instances, the higher ambiguity means that the skew is, in fact, much higher than in Senseval 2. A high skew towards the predominant sense means there are less instances from which we can learn about the rarer senses, and that we run a higher risk when labeling an instance as one of the rarer senses (instead of defaulting to the predominant one).

Senseval 2	Depend		Manual	
SVM	14.3	(60.1)	16.9	(60.4)
MaxEnt	6.3	(66.9)	17.1	(56.7)
LP	8.9	(63.3)	14.8	(49.4)

Senseval 3	Depend		Manual	
SVM	17.6	(45.0)	23.3	(60.0)
MaxEnt	8.5	(55.0)	23.7	(60.9)
LP	5.6	(60.9)	17.8	(53.5)

Table 3: Percentage of non-first instances in automatically and manually labeled training data; numbers in parentheses show the classifiers' accuracy on these instances.

Our method shares some of the principles of McCarthy et al.'s (2004) unsupervised algorithm. However, instead of focusing on detecting a single predominant sense throughout the corpus, we build a dataset that will allow us to learn about and identify all existing (prevalent) senses. Despite the fact that the first-sense heuristic is a strong baseline, and fall-back option in case of limited local information, it is not a true context-specific WSD algorithm. Any approach that ignores local context, and labels all instances with a single sense has limited effectiveness when WSD is needed in an application. Context-indifferent methods run the risk of completely mistaking the predominant sense, thereby mis-labeling most of the instances, whereas approaches that consider local context are less prone to such large-scope errors.

We further analyzed the performance of our method by examining instances labeled with senses other than the most frequent one. Table 3 shows the percentage of such instances depending on the machine learner and type of training data (automatic versus manual) being employed. It also presents the classifiers' accuracy (figures in parentheses) with regard to only the non-first senses. When trained on the automatically labeled data, our classifiers tend to be more conservative in assigning non-first senses. Interestingly, we obtain similar accuracies with the classifiers trained on the manually labeled data, even though the latter assign more non-first senses. It is worth noting that the SVM labels two to three times as many instances with non-first-sense labels, yet achieves similar levels of overall accuracy to the other clas-

sifiers (compare Tables 2 and 3) and only slightly lower accuracy on the non-first senses. This would make it a better choice when it is important to have more data on rarer senses.

6 Conclusions and Future Work

We have presented an unsupervised approach to WSD which retains many of the advantages of supervised methods, while being free of the costly requirement for human annotation. We demonstrate that classifiers trained using our method can out-perform state-of-the-art unsupervised methods, and approach the accuracy of fully-supervised methods trained on manually-labeled data.

In the future we plan to scale our system to the all-words task. There is nothing inherent in our method that restricts us to the lexical sample, which we chose primarily to assess the feasibility of our ideas. Another interesting direction concerns the use of our method in a semi-supervised setting. For example, we could automatically acquire labeled instances for words whose senses are rare in a manually tagged dataset. Finally, we could potentially improve accuracy, at the expense of coverage, by estimating confidence scores on the classifiers' predictions, and assigning labels only to instances with high confidence.

Acknowledgments

The authors acknowledge the support of EPSRC (grant EP/C538447/1) and would like to thank David Talbot for his insightful suggestions.

References

S. Banerjee, T. Pedersen. 2003. Extended gloss overlaps as a measure of semantic relatedness. In *Proc. of the 18th IJCAI*, 805–810, Acapulco.

R. Barzilay, M. Elhadad. 1997. Using lexical chains for text summarization. In *Proc. of the Intelligent Scalable Text Summarization Workshop*, Madrid, Spain.

T. Briscoe, J. Carroll. 2002. Robust accurate statistical annotation of general text. In *Proc. of the 3rd LREC*, 1499–1504, Las Palmas, Gran Canaria.

A. Budanitsky, G. Hirst. 2001. Semantic distance in WordNet: An experimental, application-oriented evaluation of five measures. In *Proc. of the ACL Worskhop on WordNet and Other Lexical Resources*, Pittsburgh, PA.

Y. S. Chan, H. T. Ng. 2007. Word sense disambiguation improves statistical machine translation. In *Proc. of the 45th ACL*, 33–40, Prague, Czech Republic.

H. Daumé III. 2004. Notes on CG and LM-BFGS optimization of logistic regression.

P. Edmonds. 2000. Designing a task for SENSEVAL-2, 2000. Technical note.

C. Fellbaum, ed. 1998. *WordNet: An Electronic Database*. MIT Press, Cambridge, MA.

W. Gale, K. Church, D. Yarowsky. 1992. A method for disambiguating word senses in a large corpus. *Computers and the Humanities*, 26(2):415–439.

M. Galley, K. McKeown. 2003. Improving word sense disambiguation in lexical chaining. In *Proc. of the 18th IJCAI*, 1486–1488, Acapulco.

C. Hsu, C. Lin. 2001. A comparison of methods for multiclass support vector machines, 2001. Technical report, Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan.

B. J. Jansen, A. Spink, A. Pfaff. 2000. Linguistic aspects of web queries, 2000. American Society of Information Science, Chicago.

C. Leacock, M. Chodorow, G. A. Miller. 1998. Using corpus statistics and wordnet relations for sense identification. *Computational Linguistics*, 24(1):147–165.

Y. K. Lee, H. T. Ng. 2002. An empirical evaluation of knowledge sources and learning algorithms for word sense disambiguation. In *Proc. of the EMNLP*, 41–48, NJ.

M. Lesk. 1986. Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In *Proc. of the 5th SIGDOC*, 24–26, New York, NY.

D. Lin. 1998. Automatic retrieval and clustering of similar words. In *Proc. of the ACL/COLING*, 768–774, Montreal.

D. McCarthy, R. Koeling, J. Weeds, J. Carroll. 2004. Finding predominant senses in untagged text. In *Proc. of the 42th ACL*, 280–287, Barcelona, Spain.

R. F. Mihalcea, P. Edmonds, eds. 2004. *Proc. of the SENSEVAL-3*, Barcelona, 2004.

R. F. Mihalcea. 2002. Word sense disambiguation with pattern learning and automatic feature selection. *Natural Language Engineering*, 8(4):343–358.

H. T. Ng, B. Wang, Y. S. Chan. 2003. Exploiting parallel texts for word sense disambiguation: an empirical study. In *Proc. of the 41st ACL*, 455–462, Sapporo, Japan.

H. T. Ng. 1997. Getting serious about word sense disambiguation. In *Proc. of the ACL SIGLEX Workshop on Tagging Text with Lexical Semantics: Why, What, and How?*, 1–7, Washington, DC.

Z. Y. Niu, D. H. Ji, C. L. Tan. 2005. Word sense disambiguation using label propagation based semi-supervised learning. In *Proc. of the 43rd ACL*, 395–402, Ann Arbor.

J. Preiss, D. Yarowsky, eds. 2001. *Proc. of the 2nd International Workshop on Evaluating Word Sense Disambiguation Systems*, Toulouse, France, 2001.

G. Ramakrishnan, A. Jadhav, A. Joshi, S. Chakrabarti, P. Bhattacharyya. 2003. Question answering via Bayesian inference on lexical relations. In *Proc. of the ACL 2003 workshop on Multilingual summarization and QA*, 1–10.

R. Snow, S. Prakash, D. Jurafsky, A. Y. Ng. 2007. Learning to merge word senses. In *Proc. of the EMNLP/CoNLL*, 1005–1014, Prague, Czech Republic.

J. Weeds. 2003. *Measures and Applications of Lexical Distributional Similarity*. Ph.D. thesis, University of Sussex.

D. Yarowsky. 1992. Word-sense disambiguation using statistical models of Roget's categories trained on large corpora. In *Proc. of the 14th COLING*, 454–460, Nantes, France.

X. Zhu, Z. Ghahramani. 2002. Learning from labeled and unlabeled data with label propagation. Technical report, CMU-CALD-02, 2002.